Decoding the Golay Code

E. R. Berlekamp[1]
Communications Systems Research Section

A procedure is described for correcting all patterns of three or fewer errors with the (23, 12) or (24, 12) Golay code. The procedure decodes any 24-bit word in about 26 "steps," each of which consists of only a few simple operations such as counting the number of ones in a 12-bit word. The procedure is based on the circulant viewpoint introduced by Karlin (1969). In addition it is shown how the (24, 12) Golay code can be used to correct certain patterns of more than three errors.

I. Introduction

Recently there has been a revival of interest in the use of binary block codes for deep space telemetry, since such codes can be used as the "outer" codes in concatenation schemes. These concatenation schemes are an attractive method of providing the very low bit error probabilities which will be required for the nonvideo science experiments on future deep space missions.

One of the most powerful known block codes is the Golay (24, 12) code, which is known to be capable of correcting all patterns of three or fewer bit errors. In Section II we describe a simple method of actually correcting these errors; this makes the Golay code (perhaps interleaved enough to deal with the bursts caused by the "inner" channel) a very attractive candidate for the "outer" code in certain concatenation schemes. In Section III we show how the Golay code can be used to correct certain patterns of more than three errors.

[1]Consultant, Department of Mathematics and Electrical Engineering, University of California, Berkeley.

II. The Algorithm

It is known that the parity-check matrix of the (24, 12) Golay code may be written as

$$H = \left[\begin{array}{c|c} I & A \end{array}\right]$$

where I is the 12×12 identity matrix and

$$A = \left[\begin{array}{l} 11011100010\ 1 \\ 01101110001\ 1 \\ 10110111000\ 1 \\ 01011011100\ 1 \\ 00101101110\ 1 \\ 00010110111\ 1 \\ 10001011011\ 1 \\ 11000101101\ 1 \\ 11100010110\ 1 \\ 01110001011\ 1 \\ 10111000101\ 1 \\ \\ 11111111111\ 0 \end{array}\right]$$

Let $\hat{A}$ denote the 11×11 upper left submatrix of A. $\hat{A}$ is a circulant matrix, each row of which is obtained by a cyclic right shift of the previous row. If the rows and columns of $\hat{A}$ are labeled from 0 to 10, then

$$\hat{A}_{i,j} = \begin{cases} 1 \text{ if } j - i \text{ is 0 or a quadratic residue modulo 11} \\ 0 \text{ if } j - i \text{ is a quadratic nonresidue modulo 11} \end{cases}$$

From this, it is easily seen that $A^{-1} = A^t$.

Each codeword of the (24, 12) code may be written as a row vector $\mathbf{C}$, which satisfies the equation $H\mathbf{C}^t = \mathbf{0}$. If $\mathbf{C}$ is transmitted and $\mathbf{R}$ is received, then the channel error pattern is $\mathbf{E} = \mathbf{R} - \mathbf{C}$. The syndrome of $\mathbf{R}$ is the 12-dimensional column vector $\mathbf{s}^t$ defined by

$$\mathbf{s}^t = H\mathbf{R}^t$$

Since $H\mathbf{E}^t = H\mathbf{R}^t - H\mathbf{C}^t = H\mathbf{R}^t$, the syndrome of the received word is the same as the syndrome of the error word, and this is the sum of the columns of the H matrix corresponding to the error locations.

Let $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_{12}$ denote the 12 unit row vectors in 12 dimensions (e.g., $\mathbf{u}_3$ =[001000000000]); let $\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_{12}$ denote the rows of A, so that

$$A = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{A}_{12} \end{bmatrix}$$

and let $A_{13}^t, A_{14}^t, \cdots, A_{24}^t$ denote the columns of A, so that

$$A = [\mathbf{A}_{13}^t | \mathbf{A}_{14}^t | \cdots | \mathbf{A}_{24}^t]$$

The syndrome $\mathbf{s}^t = H\mathbf{E}^t$ may now be represented as

$$\mathbf{s} = \sum_{i=1}^{12} E_i \mathbf{u}_i + \sum_{i=13}^{24} E_i \mathbf{A}_i$$

Similarly,

$$A^t\mathbf{s} = A^tH\mathbf{E} = [A^t | I] \cdot \mathbf{E}$$

whence

$$A^t\mathbf{s} = \sum_{i=1}^{12} E_i \mathbf{A}_i + \sum_{i=13}^{24} E_i \mathbf{u}_{i-12}$$

If we now assume that $|\mathbf{E}| \leq 3$, then at least one of the following must be true:

Case I:

$$|[E_{13}, E_{14}, \cdots, E_{24}]| = 0, \qquad |\mathbf{s}| \leq 3, \qquad \sum_{i=1}^{12} E_i \mathbf{u}_i = \mathbf{s}$$

Case II:

$|[E_{13}, E_{14}, \cdots, E_{24}]| = 1$, there exists a_j, $13 \leq j \leq 24$ for which

$$|\mathbf{s} + \mathbf{A}_j| \leq 2, \qquad \sum_{i=1}^{12} E_i \mathbf{u}_i = \mathbf{s} + \mathbf{A}_j$$

Case III:

$$|[E_1, E_2, \cdots, E_{12}]| = 0, \qquad |\mathbf{s}A| = |A^t\mathbf{s}^t| \leq 3,$$

$$\sum_{i=13}^{24} E_i u_{i-12} = \mathbf{s}A$$

Case IV:

$|[E_1, E_2, \cdots, E_{12}]| = 1$, there exists a_j, $1 \leq j \leq 12$ for which

$$|\mathbf{s}A + \mathbf{A}_j| \leq 2, \qquad \sum_{i=13}^{24} E_i \mathbf{u}_{i-12} = \mathbf{s}A + \mathbf{A}_j$$

Hence, the decoding can be accomplished simply by weighing each of these 26 vectors:

$$\mathbf{s}, \mathbf{s} + \mathbf{A}_1, \mathbf{s} + \mathbf{A}_2, \cdots, \mathbf{s} + \mathbf{A}_{12},$$
$$\mathbf{s}A, \mathbf{s}A + \mathbf{A}_1, \mathbf{s}A + \mathbf{A}_2, \cdots, \mathbf{s}A + \mathbf{A}_{12}$$

For example, suppose $\mathbf{s}$ = 100011010010. Since $|\mathbf{s}| > 3$, we compute $\mathbf{s} + \mathbf{A}_1 = \mathbf{s}$ + 110111000101 = 010100010111. Since $|\mathbf{s}+\mathbf{A}_1| > 2$, we compute $|\mathbf{s}+\mathbf{A}_2| = 6 > 2$, $|\mathbf{s}+\mathbf{A}_3| = 6 > 2$, $|\mathbf{s} + \mathbf{A}_4| = 8 > 2$, $|\mathbf{s} + \mathbf{A}_5| = 6 > 2$, $|\mathbf{s} + \mathbf{A}_6| = 8 > 2$, $|\mathbf{s} + \mathbf{A}_7| = 4 > 2$, $|\mathbf{s} + \mathbf{A}_8| = 4 > 2$, $|\mathbf{s} + \mathbf{A}_9| = 10 > 2$, $|\mathbf{s}+\mathbf{A}_{10}| = 8 > 2$, $|\mathbf{s}+\mathbf{A}_{11}| = 6 > 2$, $|\mathbf{s}+\mathbf{A}_{12}| = 6 > 2$. It is now clear that if $|E| \leq 3$, then $|[E_1, E_2, \cdots, E_{12}]| > 1$ and hence $|[E_{13}, E_{14}, \cdots, E_{24}]| \leq 1$. So we continue by computing $A^t\mathbf{s}^t = \mathbf{s}A$ = 100110100111, $|\mathbf{s}A| = 7 > 3$, $\mathbf{s}A + \mathbf{A}_1$ = 010001100010, $|\mathbf{s}A + \mathbf{A}_1| = 4 > 2$, $|\mathbf{s}A + \mathbf{A}_2| = 6 > 2$, $|\mathbf{s}A + \mathbf{A}_3| = 6 > 2$, $|\mathbf{s}A + \mathbf{A}_4| = 6 > 2$, $|\mathbf{s}A + \mathbf{A}_5| = 8 > 2$, $|\mathbf{s}A + \mathbf{A}_6| = 4 > 2$, $\mathbf{s}A + \mathbf{A}_7$ = 000100010000. Since $|\mathbf{s}A + \mathbf{A}_7| = 2$, $E_7 = 1$ and $\mathbf{E}$ = 000000100000000100010000.

Most of the decoding effort is counting the weights of the 26 relevant 12-bit vectors. For this reason, this decoding algorithm is particularly well-suited to computers

which have this instruction built in, such as the CDC 6400, 6500, 6600, and 7600. If programmed on a machine which is unable to count the weight of a 12-bit word in a single instruction, the easiest way to obtain this quantity is usually to break up the 12-bit word into pieces (say 2 pieces of 6 bits each or 3 pieces of 4 bits each) and obtain the weight of each piece by looking it up in a table.

III. Decoding More Than Three Errors

The Golay code has 2^{12} codewords of length 23, and since $\binom{23}{0} + \binom{23}{1} + \binom{23}{2} + \binom{23}{3} = 2^{23-12}$, every coset contains one word of weight ≤ 3. However, the extended Golay code, which has 2^{12} codewords of length 24 has $\binom{24}{1} + \binom{24}{3} = 2^{11}$ cosets of odd weight and another $2^{11} = \binom{24}{0} + \binom{24}{2} + \frac{1}{6}\binom{24}{4}$ cosets of even weight. It is thus possible to correct $\frac{1}{6}$ of the $\binom{24}{4}$ possible error patterns of weight 4. Some of these words of weight 4 correspond to short bursts. Even though the space channel itself is memoryless, the convolutional code will occasionally make mistakes which the Golay code will see as error bursts. For this reason, short bursts of weight 4 are more probable error patterns than long bursts of weight 4.

The sum of any two words of weight 4 from the same coset is a codeword of weight 8. Hence we may gain a considerable amount of information about which words of weight 4 are correctable and which are not by studying the codewords of weight 8.

Since there is exactly one codeword of weight 8 which has 1s in any given five positions, the total number of codewords of weight 8 is $24 \times 23 \times 22 \times 21 \times 20/8 \times 7 \times 6 \times 5 \times 4 = 3 \times 11 \times 23$. Each codeword of weight 8 has 23 distinct cyclic shifts. The codewords of weight 8 lie in 33 sets of 23 codewords each. Furthermore, each codeword of weight 8 can be mapped into 11 different codewords by the permutation $C(x) \rightarrow C(x^{2^i}) \bmod (x^{23} + 1)$, for $i = 0, 1, \cdots, 10$. Under this permutation, there are only 3 equivalence classes of codewords. The 11 members of each class are listed in Table 1.

The most probable error patterns of weight 4 are those which are due to the sum of one or more short bursts. The solid burst of length 5 occurs in codeword number 23 of Table 1. By inspecting this word, we see that the solid burst of length 5 in positions 0, 1, 2, 3, 4 lies in the same coset as the pattern of three isolated errors in positions 7, 10, and 12. Hence, if all error patterns of weight ≤ 3 are corrected, then a solid burst of length 5 cannot be corrected.

There are five codewords which contain solid bursts in positions 0, 1, 2, 3. These words may be found as cyclic shifts of lines numbered 1, 22, 23, 23, and 26 of Table 1. Since no codeword of weight 8 contains two disjoint solid bursts of length 4, *all solid bursts of length 4 may be corrected by the extended Golay code of length 24.*

A burst of length 5 and weight 4 must be of one of the following three types: 11101, 11011, 10111. Type 11101 is contained in Table 1 codewords numbered 2, 6, 22, 23, 24, Type 11011 in codewords numbered 1, 6, 11, 14, 23, and Type 10111 in codewords numbered 1, 2, 5, 12, 23.

The most probable error patterns of weight four are those which are due to the sum of one or two short bursts. These types of error patterns and the codewords of Table 1 which contain them are as follows:

Error type	Reference numbers of Table 1 codewords
111 plus 1	1, 3, 4, 5, 6, 11, 12, 17 21, 22, 23, 24, 26, 28, 33
11 plus 11	1, 3, 4, 6, 8, 9, 10, 11, 12, 14 17, 21, 25, 26, 27, 29, 30, 31, 32, 33

An examination of the conflicts between the goal of correcting 111 plus 1 and 11 plus 11 reveals the following dangerous codewords through positions 0, 1, 2:

Codeword	Reference number
0, 1, 2 7, 8, 9, (13), (17)	4
0, 1, 2, (6), (10) 16, 17, 18	4
0, 1, 2, 4, 5, (9), 18, 19	6
0, 1, 2, 10, 11, 13, 14, (19)	11
0, 1, 2, 5, 6, 12, 13, (15)	33

This shows that any pair of two sets of double adjacent errors can be corrected and that one can also correct any error pattern of the type 111 plus 1 (i.e., a solid burst of length 3 and an additional isolated error) unless the isolated error follows the burst by 6, 9, 10, 13, 17, 15, or 19 digits. If the isolated error follows the burst of length 3 by 9, 19, or 15, the syndrome is the same as for a pair of bursts of length 2; if the isolated error follows the burst of 3 by 13, 17, 6, or 10, then there is ambiguity with another error pattern of the same type.

Bibliography

Berlekamp, E. R., *Algebraic Coding Theory,* McGraw-Hill Book Co., New York, 1968 (see Sec. 15.2, pp. 352–361, particularly Eq. 15.242, p. 357).

Karlin, M., "New Binary Coding Results by Circulants," *IEEE Trans. Inform. Th.*, Vol. 15, pp. 81–92 (see Fig. 1, p. 82).

Table 1. One codeword of weight 8 from each of the 33 cyclic equivalence classes

Reference number	Positions of 1s in the codeword								Lengths (origins) of solid bursts
1	0	8	12	13	7	5	10	11	4(10) 2(7)
2	0	16	1	3	14	10	20	22	3(22)
3	0	9	2	6	5	20	17	21	2(5) 2(20)
4	0	18	4	12	10	17	11	19	3(10) 3(17)
5	0	13	8	1	20	11	22	15	3(22)
6	0	3	16	2	17	22	21	7	3(21) 2(2) 2(16)
7	0	6	9	4	11	21	19	14	
8	0	12	18	8	22	19	15	5	2(18) 2(22)
9	0	1	13	16	21	15	7	10	2(0) 2(15)
10	0	2	3	9	19	7	14	20	2(2) 2(19)
11	0	4	6	18	15	14	5	17	3(4) 2(14) 2(17)
12	0	2	4	6	5	10	11	∞	3(4) 2(10)
13	0	4	8	12	10	20	22	∞	2(22)
14	0	8	16	1	20	17	21	∞	2(16) 2(20) 2(0)
15	0	16	9	2	17	11	19	∞	2(16)
16	0	9	18	4	11	22	15	∞	2(22)
17	0	18	13	8	22	21	7	∞	3(21) 2(7)
18	0	13	3	16	21	19	14	∞	2(13)
19	0	3	6	9	19	15	5	∞	2(5)
20	0	6	12	18	15	7	10	∞	2(6)
21	0	12	1	13	7	14	20	∞	3(12) 2(0)
22	0	1	2	3	14	5	17	∞	4(0)
23	0	1	2	4	3	12	7	10	5(0)
24	0	2	4	8	6	1	14	20	3(0)
25	0	4	8	16	12	2	5	17	2(4) 2(16)
26	0	8	16	9	1	4	10	11	4(8) 2(0)
27	0	16	9	18	2	8	20	22	2(8) 2(22)
28	0	9	18	13	4	16	17	21	3(16)
29	0	18	13	3	8	9	11	19	2(8) 2(18)
30	0	13	3	6	16	18	22	15	2(15) 2(22)
31	0	3	6	12	9	13	21	7	2(6) 2(12)
32	0	6	12	1	18	3	19	14	2(0) 2(18)
33	0	12	1	2	13	6	15	5	3(0) 2(5) 2(12)